Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Image-based traffic signal control via world models*

Xingyuan DAI[1,2], Chen ZHAO[1,2], Xiao WANG[3], Yisheng LV[1,2], Yilun LIN[4], Fei-Yue WANG[†‡1,2]

[1]*The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China*

[2]*School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China*

[3]*School of Artificial Intelligence, Anhui University, Hefei 230039, China*

[4]*Shanghai AI Laboratory, Shanghai 200232, China*

[†]E-mail: feiyue.wang@ia.ac.cn

Received July 28, 2022; Revision accepted Oct. 6, 2022; Crosschecked Nov. 4, 2022

Abstract: Traffic signal control is shifting from passive control to proactive control, which enables the controller to direct current traffic flow to reach its expected destinations. To this end, an effective prediction model is needed for signal controllers. What to predict, how to predict, and how to leverage the prediction for control policy optimization are critical problems for proactive traffic signal control. In this paper, we use an image that contains vehicle positions to describe intersection traffic states. Then, inspired by a model-based reinforcement learning method, DreamerV2, we introduce a novel learning-based traffic world model. The traffic world model that describes traffic dynamics in image form is used as an abstract alternative to the traffic environment to generate multi-step planning data for control policy optimization. In the execution phase, the optimized traffic controller directly outputs actions in real time based on abstract representations of traffic states, and the world model can also predict the impact of different control behaviors on future traffic conditions. Experimental results indicate that the traffic world model enables the optimized real-time control policy to outperform common baselines, and the model achieves accurate image-based prediction, showing promising applications in futuristic traffic signal control.

Key words: Traffic signal control; Traffic prediction; Traffic world model; Reinforcement learning
https://doi.org/10.1631/FITEE.2200323 **CLC number:** U491; TP181

1 Introduction

Traffic signal control plays a pivotal role in relieving urban congestion, a critical issue in social, economic, and environmental concerns worldwide (Wang FY, 2010). With the development of sensing, communication, and intelligence technologies, traffic signal control is experiencing a transformation from fixed-time control to passive control to proactive control (Mei et al., 2019). A fixed-time controller phase sequence and duration are pre-determined by professional engineers using their experience or rules. Because the controller lacks perception, it cannot flexibly adapt to traffic flows with varying patterns. Passive signal control, including classic actuated control (Newell, 1969) and model-free data-driven control (Li L et al., 2016), enables controllers to react to traffic conditions observed by associated sensors. However, the controller responses tend to be slower than changes in traffic status, leading to suboptimal control policies. By predicting traffic status, proactive signal control can adjust the timing plan in advance and, ideally, make traffic flow change according to the controller expectation.

Many proactive control system solutions have been proposed to incorporate prediction modules to facilitate traffic control, and parallel transportation

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (Nos. 62173329 and U1811463)
ORCID: Xingyuan DAI, https://orcid.org/0000-0001-7517-5049; Fei-Yue WANG, https://orcid.org/0000-0001-9185-3989

systems (Wang FY, 2010; Xiong et al., 2020) are the representation. Parallel transportation systems integrate actual and corresponding artificial transportation systems. The artificial ones can be used to optimize the control policy through computational experiments and prescribe actual system operations through parallel execution. An effective prediction model plays a vital role in the two processes. In the optimizing process, the prediction model introduces dynamic environment descriptions to improve data efficiency and help achieve near-optimal control (Li L et al., 2017). In the prescription process, the prediction model helps human beings interpret decisions made by artificial systems in human-in-the-loop systems, like parallel recommendation systems (Zhao et al., 2017; Zhu et al., 2020; Jin et al., 2021). In other words, if control agents cannot predict the potential effect of recommended policies or provide reasonable explanations, signal control engineers may not execute the procedure.

To better integrate traffic prediction and control in proactive control systems and to explore efficient traffic modeling paradigms for artificial systems of parallel transportation systems, we need first to consider the following questions:

1. What kind of traffic state formation can comprehensively describe the complex traffic conditions that encompass the interaction and evolution of vehicle entities in intersections?

2. How should we cope with complex traffic characteristics and learn an efficient traffic model to capture traffic dynamics while performing accurate multi-step predictions?

3. How can we achieve effective and real-time decision-making by combining the prediction model with traffic control to meet the needs of traffic management systems for robustness, predictability, and timeliness?

These three problems are also of interest in model-based traffic signal control, but most classical studies do not thoroughly answer these questions, especially regarding tradeoffs between effectiveness and timeliness. Specifically, classical planning-based traffic signal control (Guo et al., 2019) and model predictive control (MPC) (Hao et al., 2018; Ye et al., 2019) usually use mathematical models to describe lane-level traffic dynamics, such as the changes in vehicle trajectories, queue length, and density. These approaches perform online planning for optimal control in the execution stage, which may lead to difficulties in real-time decision-making (Ye et al., 2019). Classical model-based reinforcement learning (RL), which refers mainly to tabular dynamic programming (Wiering, 2000), constructs lookup tables for traffic prediction and value functions via data-driven methods, but the tabular form cannot comprehensively describe traffic dynamics.

Model-based deep reinforcement learning (DRL) (Wang HN et al., 2020), with its advantages in representation learning and dynamics learning, promises to answer the three questions. However, most studies about model-based DRL (Yu et al., 2020; Zhang HC et al., 2020) focus mainly on improving control performance based on lane-level traffic states (e.g., the queue length, density, and traffic flow). These works have not explored a way to build a multi-step prediction model that comprehensively describes traffic dynamics and provides a horizon for controllers. In addition, some recent works incorporate traffic flow prediction (Lv et al., 2014) in a DRL-based traffic control framework (Kim and Jeong, 2019; Abdoos and Bazzan, 2021), but the traffic flow prediction model that forecasts 5- to 60-min future traffic volumes is unable to help controllers deal with near-future traffic conditions.

This study provides a feasible solution for the three problems using model-based DRL, and selects an isolated traffic signal control scenario to detail the method. To answer the first question, we formalize the intersection traffic states as images that contain vehicle positions, and use image streaming to describe traffic dynamics comprehensively. The intersection image, which can be obtained from cameras and vehicular networks (Liang et al., 2019; Wang J et al., 2020), possesses more information than lane-level traffic state vectors, such as the intersection topology and semantic context associated with traffic entities.

To answer the last two questions, we introduce a traffic world model to perform multi-step traffic prediction and facilitate the learning of effective and real-time control policies. The traffic world model learned via a model-based DRL approach, DreamerV2 (Hafner et al., 2022), excels in capturing the spatiotemporal feature in image-based traffic dynamics. Specifically, the model introduces a compact latent space corresponding to the original traffic image space and can perform predictions in the latent

space. The latent state representation, concatenating deterministic and stochastic latent variables, is suitable for capturing fixed patterns and randomness in traffic environments. In the policy optimization stage, based on informative and compact state representation, the traffic world model can generate many samples through latent predictions to facilitate policy optimization without access to high-dimensional traffic images. In the execution stage, traffic signal controllers can make real-time decisions based on latent representations of historical and current traffic states without online planning like MPC.

The main contributions of this study can be summarized as follows:

1. We introduce a new paradigm, a traffic world model, for image-based real-time traffic signal control. Unlike MPC, the world model decouples prediction and decision modules to balance effectiveness and timeliness for traffic signal control. The world model enables image-based multi-step traffic prediction to visualize the impact of control behaviors on future traffic conditions.

2. We implement the traffic world model using DreamerV2 and leverage its informative compact latent space to substitute high-dimensional image space to realize efficient exploration for a high-performance traffic control policy.

3. Empirical results show that the control policy optimized via the world model achieves better control performance than baselines, and requires shorter decision time than MPC approaches while maintaining accurate multi-step prediction for image-based traffic dynamics.

2 Traffic signal control via images

In this section, the traffic signal control problem is formulated as an image-based Markov decision process (MDP), which will be tackled by DreamerV2 with the world model introduced in Section 3. We focus on quantifying traffic states as an image form and explaining the motivation.

2.1 Markov decision process settings

An MDP describes a sequential decision-making process: At each time step, the agent performs an action based on its policy and the environment state; then, the environment returns a reward as an evaluation for the action and a subsequent state according to its dynamics model. A typical MDP is formed by a quintuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, $\mathcal{P}\colon \mathcal{S} \times \mathcal{A} \mapsto \mathcal{S}$ is the transition probability from any state s ($s \in \mathcal{S}$) to any state s' ($s' \in \mathcal{S}$) given action a ($a \in \mathcal{A}$), $\mathcal{R}\colon \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the reward function that determines immediate rewards received by the agent for a transition from (s, a) to s', and γ is the discount factor used for constructing the return, i.e., long-term rewards.

In the traffic signal control problem, the signal controller at the intersection can be viewed as an agent, and the control period is denoted as Δt, representing the time interval in the environment between two consecutive time steps. At each time step t, the agent observes the state s_t from the traffic environment, and executes an action $a_t = \pi(s_t)$ for signal control based on its policy π and state s_t. The traffic environment then returns the next state s_{t+1} according to the traffic model $p(s_{t+1}|s_t, a_t)$ and a reward r_t that evaluates the control performance last time.

2.2 State design and motivation

The state s_t ($s_t \in \mathcal{S}$) denotes the environment's condition at time step t available to the agent and can be used for decision-making and prediction in model-based DRL. This study selects quantified top-view images that contain vehicle positions at the intersection as states. Fig. 1a shows a diagram of the quantifying process for the position image. We first divide the whole intersection into small grids. Then, we assign a value to each grid; the value of each grid is a scalar indicating the number of vehicles whose center point is in the grid. Finally, we normalize the quantified image by dividing the number of vehicles in each grid by the maximum number of vehicles in all grids. It should be pointed out that there are two types of grid lengths and widths, to ensure that vehicles in different lanes are distinguished in the quantified image. For example, as shown in Fig. 1a, the width of the grid located in the N-S lanes, denoted as $w_{\text{lane}}^{\text{grid}}$, is equal to the lane width w_{lane}. The width of the grid outside the N-S lanes is $w_{\text{out}}^{\text{grid}}$. Its value can be calculated if we know the width of the study area w_i, number of lanes $n_{\text{lane}}^{\text{grid}}$, width of the quantified image n_i^{grid}, and lane width w_{lane}.

In this study, we formalize the intersection traffic state at each time step as a 64×64 position image (Fig. 1b). In practice, we have many ways to

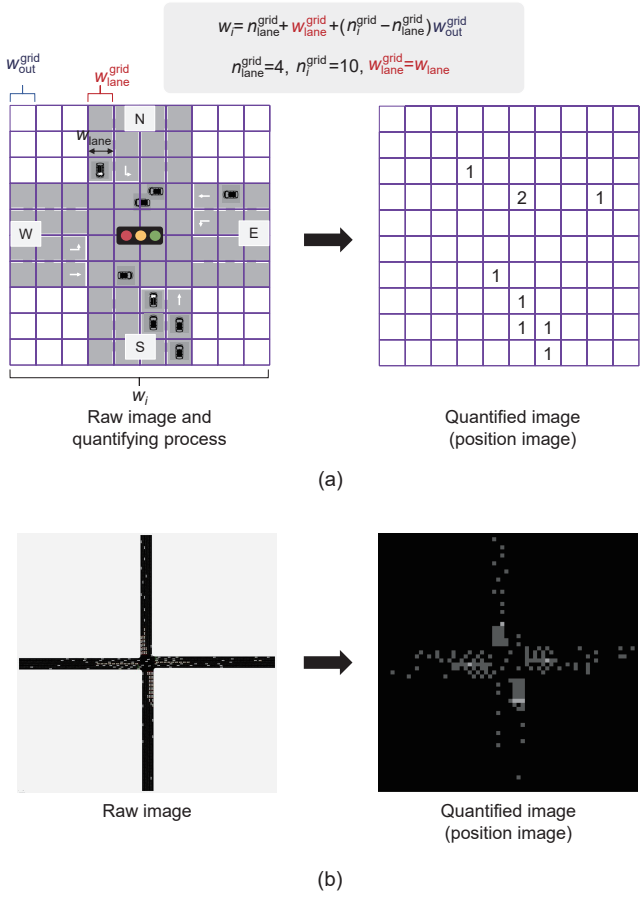

Fig. 1 State formation for the intersection: (a) diagram of the quantifying process and the quantified position image; (b) quantified position image of the intersection studied in this study

obtain the position image. We can form the image by quantifying the raw snapshots from traffic video surveillance systems or obtaining vehicles' positions directly from vehicular networks (Liang et al., 2019). This study focuses mainly on explaining the idea of image-based traffic signal control and formalizes traffic states as 64×64 quantified images. However, the applied model-based DRL framework could handle more complex state formation, like raw images obtained from cameras or larger quantified images that reflect more fine-grained traffic conditions.

The motivation for using images as traffic states is derived from the image's broad application potential for traffic signal control. First, multi-frame raw images of the intersection contain sufficient and diverse traffic information: lane-level states like the queue length, density, and vehicle speed; fine-grained semantics like the road network topology and traffic entity relationship; natural information like weather and illumination. Ideally, an effective traffic signal controller can capture the relationship between the road structure and vehicles. However, typical traffic signal control methods consider mainly lane-level traffic states, which restricts the controllers to dealing with more difficult traffic conditions beyond lanes, like intersection emergencies, unusual road network topologies, and mixed vehicle-pedestrian scenes. Second, expert signal control engineers use intersection images from surveillance systems to adjust traffic signal control systems (Jin et al., 2021). So, if we want to train artificial signal control engineers that can precisely imitate real engineers' behavior in parallel transportation systems, a direct approach is to feed artificial ones with the same images that real ones use. Third, images are scalable and can be fused with other forms of data like lane-level data or a point cloud to help achieve better control performance. The field may also benefit from multimodal

fusion, a hot research topic for autonomous driving (Nie et al., 2021; Xiao et al., 2022).

Although utilization of image-based traffic states has various advantages, how to leverage the state with high dimensions for efficient traffic signal control is still a complex problem. We provide a solution using model-based DRL in Section 3.

2.3 Action and reward

The agent takes action a_t ($a_t \in \mathcal{A}$) at time step t based on its policy and the environment's state. Considering the flexibility, we choose phase selection as the action for traffic signal control. Specifically, a feasible phase set is predefined, and the agent selects a phase from the set every Δt. In the next step, the signal controller executes the action if the current green phase duration is between the minimum $t^{\mathrm{g}}_{\min}$ and maximum $t^{\mathrm{g}}_{\max}$. Otherwise, the controller will keep the current phase or switch to the next phase according to the predefined phase sequence. Moreover, there is a yellow interval t_{y} ($t_{\mathrm{y}} < \Delta t$) if two consecutive executed phases are different.

The reward r_t is the immediate feedback from the environment through the reward function $r(s_t, a_t)$, indicating how good the action a_t is for the state s_t. We select pressure (Varaiya, 2013) as the reward. Pressure is a notion in traffic theory that indicates network-level stability by considering intersection-level traffic properties (Varaiya, 2013). The pressure for an intersection is defined as inflow minus outflow. It has been analytically proven that minimizing pressure is equivalent to minimizing the average travel time and maximizing network throughput (Wei et al., 2019b), which are both challenging to optimize directly as rewards. The reward in this study is defined as

$$r_t = \sum_{o \in O} v^o_{t+1} - \sum_{l \in L} v^l_{t+1}, \tag{1}$$

where L and O denote the sets of incoming lanes and outgoing lanes of the intersection, respectively. The number of vehicles at time step $t+1$ is denoted as v_{t+1}.

2.4 Image-based traffic model

A standard dynamics model $p(s_{t+1}|s_t, a_t)$ describes the transition probability from current state s_t to the next state s_{t+1} given the action a_t (Sutton and Barto, 2018). This study considers quantified intersection images as states, and the traffic model reveals the relationship between vehicle position changes and signal controller actions in the intersection. From the microscopic traffic model, we can obtain comprehensive traffic information at the intersection, like vehicle trajectories, and changes in traffic flow or queue length.

Unlike image-based traffic models, lane-level traffic models that are adopted in classical planning-based control, MPC, and most data-driven traffic signal control methods, have difficulty in describing comprehensive traffic information. Traditional mathematical macroscopic traffic models like METANET and cell transmission models (Ye et al., 2019) cannot accurately describe traffic micro-variations; data-driven traffic models that consider changes in lane-level traffic flows or vehicle speeds focus only on particular parts of traffic dynamics and fail to describe the vehicle in the middle of the intersection.

The image-based model contains more information than the land-level model, such as road topology, traffic entity relationships at intersections, and their evolutionary processes. The model describing traffic dynamics in more detail helps traffic signal controllers handle more complex traffic scenarios. Section 3 presents the method for learning the image-based traffic model and illustrates how to use it to learn effective control policies.

2.5 Traffic signal control problem

Assuming that we focus on the performance of traffic signal controllers in a typical intersection for a period covering a total of T time steps, we can formulate traffic signal control as the following infinite MDP problem: Given a single-intersection traffic environment, we search for a signal control policy $\pi = \{\pi \colon \mathcal{S} \mapsto \mathcal{A}\}$ for the intersection agent to maximize the expected value of the cumulative reward:

$$G = \mathbb{E}\left[\sum_{t=1}^{T} r_t \mid a_t \sim \pi\left(\cdot \mid s_t\right), s_{t+1} \sim p(\cdot \mid s_t, a_t)\right]. \tag{2}$$

3 World model for image-based traffic signal control

In Section 2.2, we formalize traffic states as 64×64 images, and the high-dimensional image leads to a

considerable policy search space. In this section, we introduce a new paradigm, the traffic world model, to develop a high-performance control policy in a significant space, while learning an accurate multi-step image-based traffic prediction model.

The world-model-based traffic signal control is different from the current mainstream approaches, such as model-free RL and MPC. In the training stage, the world-model-based approach can achieve higher data efficiency to explore effective traffic control policies than model-free RL. This characteristic makes the traffic world model suitable for scaling to real-world traffic signal control, because real-world samples are expensive, especially for corner scenarios. In the execution stage, the world model can help perform real-time traffic control and on-demand prediction, allowing traffic controllers to meet the needs for robustness, predictability, and timeliness. In contrast, model-free RL is incapable of predicting traffic dynamics, and MPC has difficulty in balancing effectiveness and timeliness due to the rolling optimization mechanism.

Fig. 2 compares the world model paradigm with model-free off-policy RL and MPC paradigms for traffic signal control. The off-policy RL-based control agent usually maintains an actor that accepts traffic states from the environment and outputs signal control actions. In contrast, the world-model-based control agent maintains both an actor and a world model. The world model that abstracts environment dynamics maps the original traffic states to latent states, and the actor uses the latent states to generate signal control actions. In the training stage, world-model-based traffic control optimization contains three components that can run interleaved or in parallel: (1) learning a world model using data collected from the interaction between agents and environments; (2) optimizing control policies via latent predictions of the world model; (3) performing control in traffic environments and collecting new experience for world model learning. In this workflow, the world model introduces a compact latent space that corresponds to the original high-dimensional traffic state space. Replacing the traffic environment, the model can generate many informative samples in the latent space to help the intersection agent efficiently learn high-performance control policies. On the contrary, off-policy RL approaches require a lot of interaction data with traffic environments for policy optimization and may encounter difficulties in handling high-dimensional traffic states.

In the execution stage, the world model can help perform real-time traffic control and implement on-demand image-based traffic prediction to evaluate the performance of control policies. On-demand prediction means that prediction is decoupled from

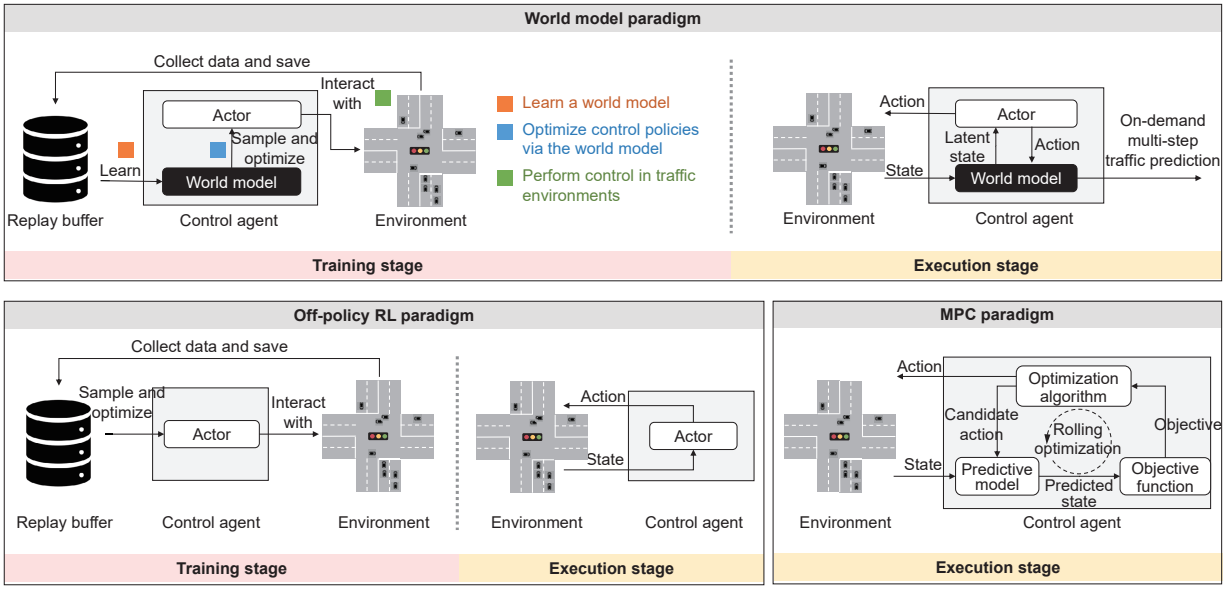

Fig. 2 Comparison of paradigms of the world model, model-free off-policy reinforcement learning (RL), and model predictive control (MPC) for traffic signal control

control. We can predict at any time the future traffic state along with currently used or other control policies. In contrast, off-policy RL approaches have no consequence for future prospects. Another widely used paradigm, MPC, which is limited by rolling optimization mechanisms, faces intractable problems for tradeoffs between effectiveness and timeliness. So, the world model paradigm is introduced to create balance.

In this study, we employ an efficient and effective model-based DRL approach DreamerV2 (Hafner et al., 2022) to build the world model. Fig. 3 details the implementation of DreamerV2 for three operations in world-model-based traffic control optimization, i.e., learning the traffic world model, optimizing control policies via the world model, and performing control in traffic environments. Note that the three operations can run interleaved or in parallel. In these operations, traffic world model learning and control policy optimization are the core of implementing accurate traffic prediction models and effective real-time signal control policies. The process of collecting new data in the third component is similar to that in general RL methods.

In the subsequent subsections, we first present the implementation of the three operations for world-model-based traffic control optimization in the training stage. Then, we introduce how to use the world model to achieve effective real-time control and on-demand accurate prediction in the execution stage.

3.1 Training stage: world model learning

In this subsection, we use DreamerV2 to learn the traffic world model from the growing dataset of the intersection agent's traffic control experience. The experience contains sequences of traffic image states $s_{1:T}$, control actions $a_{1:T}$, and rewards $r_{1:T}$. The process of collecting experience is presented in Section 3.3.

Fig. 3a shows the overall structure of the traffic world model. The world model introduces a compact latent space where the world model describes the intersection agent's experience in the traffic environment as a latent dynamics model. The latent dynamics model captures the spatiotemporal characteristics of the traffic state and replaces the traffic environment for signal control policy optimization. In the latent space, the latent state at each time step contains deterministic state h_t and random state z_t, which together encode the historical and current traffic image states, and are able to predict forward in the hidden space, reconstruct the traffic image in the original space, and return the reward.

The traffic world model consists of five modules:

$$\begin{cases} \text{Recurrent model:} \ \ h_t = f_\phi\left(h_{t-1}, z_{t-1}, a_{t-1}\right), \\ \text{Representation model:} \ \ z_t \sim q_\phi\left(z_t \mid h_t, s_t\right), \\ \text{Transition predictor:} \ \ \hat{z}_t \sim p_\phi\left(\hat{z}_t \mid h_t\right), \\ \text{Image predictor:} \ \ \hat{s}_t \sim p_\phi^s\left(\hat{s}_t \mid h_t, z_t\right), \\ \text{Reward predictor:} \ \ \hat{r}_t \sim p_\phi^r\left(\hat{r}_t \mid h_t, z_t\right). \end{cases} \tag{3}$$

The recurrent model uses the previous deterministic latent state h_{t-1}, stochastic latent state z_{t-1}, and the control action a_{t-1} to generate the current deterministic latent state h_t. The representation model first encodes the current image-based traffic state s_t as embeddings, and then uses the embeddings and the deterministic recurrent state h_t to generate the stochastic latent state z_t, which follows multiple categorical distributions. Unlike the representation model, the transition predictor uses only the deterministic latent state h_t to predict the stochastic latent state, which is called the prior latent state $\hat{z}_t$. The posterior latent state z_t fuses the current traffic state s_t via the encoder, while the prior latent state $\hat{z}_t$ tries to be close to the posterior latent state without access to the current traffic state. The image predictor reconstructs the traffic image based on the deterministic latent state h_t and the stochastic latent state z_t, and the reward predictor uses them to predict the reward.

In practice, all the modules of the world model are implemented as neural networks. Representation models use convolutional neural networks (CNNs) (Mnih et al., 2015). Recurrent models use gated recurrent units (GRUs) (Seng et al., 2021). Transition predictors and reward predictors both use multilayer perceptrons (MLPs). Image predictors use transposed CNNs to output the mean of a diagonal Gaussian with unit variance. During traffic world model learning, all modules are optimized jointly. The loss function is

$$\begin{aligned} \mathcal{L}(\phi) \doteq & \mathbb{E}_{p_\phi}\Bigg[\sum_{t=1}^{T}\Big(-\ln p_\phi^s\left(s_t \mid h_t, z_t\right) \\ & -\ln p_\phi^r\left(r_t \mid h_t, z_t\right) \\ & +\beta\,\mathrm{KL}\left[q_\phi\left(z_t \mid h_t, s_t\right) \| p_\phi\left(z_t \mid h_t\right)\right]\Big)\Bigg], \end{aligned} \tag{4}$$

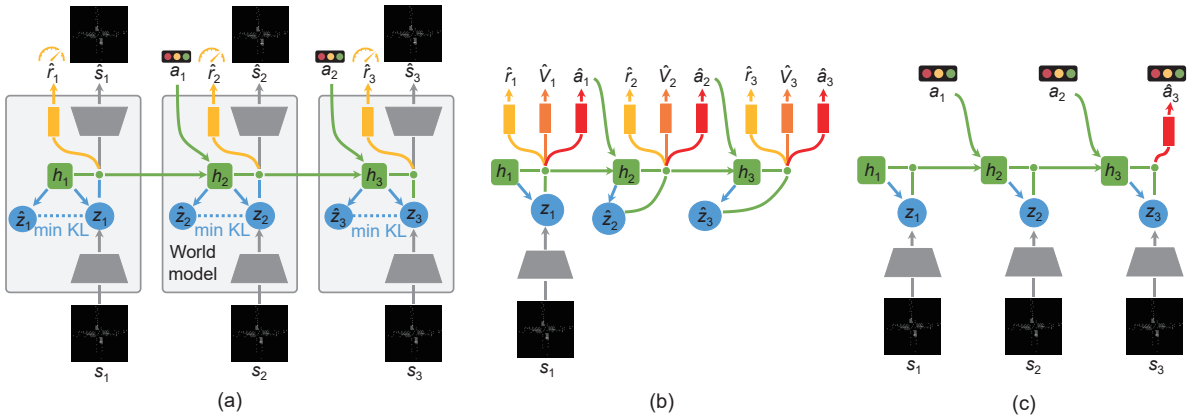

Fig. 3 Components of DreamerV2-based traffic control optimization in the training stage: (a) learning a traffic world model; (b) optimizing policies via latent predictions in the world model; (c) performing control in traffic environments

where the first item is the traffic image reconstruction loss, the second item is the reward prediction loss, and the third item is the Kullback–Leibler (KL) divergence between the posterior and prior distributions. The motivation for using the KL loss can be found in Hafner et al. (2022).

The core elements of the traffic world model are the recurrent model, representation model, and transition predictor. They form a recurrent state-space model (RSSM) (Hafner et al., 2019), also known as the latent dynamics model. With the help of the RSSM, we can emulate traffic environment operations in a compact latent space. RSSMs integrate deterministic models (e.g., recurrent neural networks) and stochastic models, which excel at capturing deterministic trends and uncertainty in traffic environments, respectively. These characteristics enable RSSMs to robustly predict traffic conditions in a compact latent space (Hafner et al., 2019).

Two properties make the traffic world model suitable for image-based traffic prediction and control. First, the world model's latent states concatenate deterministic and stochastic latent states, and the two latent states are suitable for modeling the deterministic trends and uncertain changes in traffic, respectively. The informative latent state allows agents to predict future states in the compact latent space, facilitating long-term predictions for the high-dimensional image-based traffic state. Second, the world model can efficiently predict a multitude of latent traffic state sequences in parallel in a batch, and the predicted samples can be used for policy optimization.

3.2 Training stage: traffic signal control policy optimization via the world model

The traffic world model constructs a latent space parallel to the original image space, and the intersection agent can use the latent dynamics model to optimize control policies. The decision-making process in the latent space can be viewed as a latent MDP. Here, z_t^{c} denotes the compact latent state that is the concatenation of the deterministic state h_t and the stochastic state z_t; accordingly, the estimated compact latent state $\hat{z}_t^{\mathrm{c}}$ is the concatenation of h_t and $\hat{z}_t$. In the latent MDP, the initial state $\hat{z}_0^{\mathrm{c}}$ can be obtained via world model learning. Then, given the action sequence, the transition predictor $p_\phi\left(\hat{z}_t^{\mathrm{c}} \mid \hat{z}_{t-1}^{\mathrm{c}}, \hat{a}_{t-1}\right)$ will output sequence $\hat{z}_{1:H}^{\mathrm{c}}$ of latent states up to the prediction horizon H. The mean of the reward predictor $p_\phi^r\left(\hat{r}_t \mid \hat{z}_t^{\mathrm{c}}\right)$ is used as reward sequence $\hat{r}_{1:H}$.

As shown in Fig. 3b, the control policy is optimized purely within the learned traffic world model via actor–critic learning, wherein a stochastic actor and a deterministic critic are cooperatively optimized. The actor tries to output signal control actions to maximize the critic's output, whereas the critic attempts to accurately estimate the accumulated discounted future rewards that the actor achieves. The actor and critic use the parameter

vectors ψ and ξ, respectively, and can be denoted as

$$\begin{cases} \text{Actor: } \hat{a}_t \sim p_\psi\left(\hat{a}_t \mid \hat{z}_t^{\mathrm{c}}\right), \\ \text{Critic: } v_\xi\left(\hat{z}_t^{\mathrm{c}}\right) \approx \mathbb{E}_{p_\phi, p_\psi}\left[\sum_{\tau \geq t} \gamma^{\tau-t} \hat{r}_\tau\right]. \end{cases} \tag{5}$$

The critic is updated via temporal-difference learning, and the loss function can be denoted as

$$\mathcal{L}(\xi) \doteq \mathbb{E}_{p_\phi, p_\psi}\left[\sum_{t=1}^{H-1} \frac{1}{2}\left(v_\xi\left(\hat{z}_t^{\mathrm{c}}\right) - V_t^\lambda\right)^2\right], \tag{6}$$

where the value target uses the general λ-target (Sutton and Barto, 2018), which is defined recursively as

$$V_t^\lambda \doteq \hat{r}_t + \hat{\gamma}_t \begin{cases} (1-\lambda) v_\xi\left(\hat{z}_{t+1}^{\mathrm{c}}\right) + \lambda V_{t+1}^\lambda, & \text{if } t < H, \\ v_\xi\left(\hat{z}_H^{\mathrm{c}}\right), & \text{if } t = H. \end{cases} \tag{7}$$

The actor is updated through a combined loss function:

$$\begin{aligned} \mathcal{L}(\psi) \doteq & \mathbb{E}_{p_\phi, p_\psi}\Bigg[\sum_{t=1}^{H-1}\Big(-\rho \ln p_\psi\left(\hat{a}_t \mid \hat{z}_t^{\mathrm{c}}\right) \mathrm{sg}\big(V_t^\lambda - v_\xi\left(\hat{z}_t^{\mathrm{c}}\right)\big) \\ & - (1-\rho) V_t^\lambda - \eta \mathrm{H}\left[a_t \mid \hat{z}_t^{\mathrm{c}}\right]\Big)\Bigg], \end{aligned} \tag{8}$$

where ρ and η are weighting factors, $\mathrm{sg}(\cdot)$ denotes stopping the gradients around the targets, and $\mathrm{H}[\cdot]$ denotes the entropy. The first item introduces unbiased but high-variance REINFORCE gradients (Sutton and Barto, 2018) to encourage the actor to converge to better control policies. The second item represents biased but low-variance straight-through gradients (Hafner et al., 2022) to encourage the actor to learn faster initially. The third item is an entropy bonus, which regularizes the entropy of the actor to make tradeoffs between exploration and exploitation (Hafner et al., 2022).

3.3 Training stage: traffic signal control in environments and experience collection

As shown in Fig. 3c, the intersection agent performs control actions in the traffic environment via the learned policy. In this component, the world model, actor, and critic are all fixed, and the agent's policy is determined by the encoder and latent dynamics model in the world model as well as the actor. It should be pointed out that this component does not require prediction, and the latent dynamics model is used only to encode historical states and not to predict future states. The agent interacts with the traffic environment by encoding sequences of historical traffic states from the environment and outputting control actions, and the interaction process can generate new experience to grow the dataset. The generated data containing sequences of traffic image states $s_{1:T}$, control actions $a_{1:T}$, and rewards $r_{1:T}$, will be further used to update the traffic world model as presented in Section 3.1.

3.4 Execution stage: real-time traffic signal control and performance evaluation via image-based prediction

After optimization, the learned world model can be leveraged for real-time traffic signal control and performance evaluation via image-based traffic prediction in the execution stage. Real-time traffic signal control adopts the same architecture as the third component in the training stage (Fig. 3c). The control agent encodes historical traffic states, actions, and the current traffic state as the latent state, and generates an action to perform in the environment. This decision process does not require online planning, which ensures real-time performance.

Apart from real-time traffic control, we can use the learned world model to make multi-step predictions for image-based traffic states and estimate the control performance in the execution stage. Because the on-demand prediction process is separated from the control process, it can be performed at any time. The world model enables flexible prediction of future traffic states along with the current control policy or other control policies. An example is illustrated in Fig. 4. Given the traffic states of two consecutive steps s_1 and s_2, the initial deterministic latent state h_1, and the action a_1, the intersection agent first encodes them as latent states (h_2, z_2). Then, the agent uses the model to predict future latent states $(h_3, \hat{z}_3)$ and $(h_4, \hat{z}_4)$ based on future sequence of actions a_2 and a_3 obtained from policies. Finally, the agent uses the image predictor to predict future traffic image states based on future latent states.

4 Experimental results

This section compares the control performance and decision time of world-model-based DreamerV2 policies with traditional rule-based controllers,

model-free RL, and MPC policies. We validate that DreamerV2 policies achieve real-time and robust control as well as accurate multi-step prediction.

4.1 Environment setups

We use Simulation of Urban MObility (SUMO) (Lopez et al., 2018) as the traffic environment to evaluate the performance of signal control policies. The experiments are conducted in three datasets: (1) a synthetic single-intersection dataset, $D_{1\times1}$, with four traffic patterns to model flat and peak traffic demands and cover unsaturated, saturated, and oversaturated traffic flows; (2) a synthetic two-intersection dataset, $D_{2\times1}$, with four types of vehicles to simulate the complex intersection environment; (3) a real three-intersection dataset, $D_{3\times1}$, to verify the intersection coordination ability for the centralized controller.

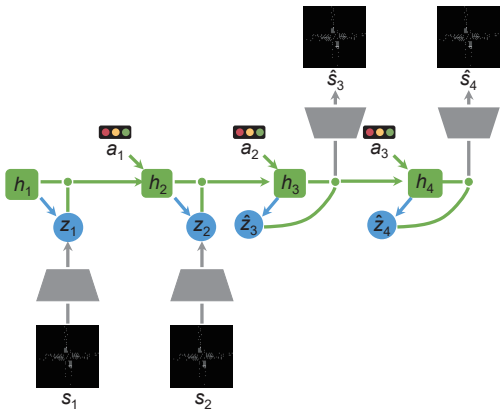

Fig. 4 Multi-step prediction using the traffic world model

The configurations for the traffic datasets are shown in Table 1. We set the environment to last 600 s and the control period Δt to be 5 s. The minimum duration $t^{\mathrm{g}}_{\mathrm{min}}$ and maximum duration $t^{\mathrm{g}}_{\mathrm{max}}$ of green phases are set at 5 s and 60 s respectively, and the yellow time t_{y} is set at 3 s. During policy evaluation, each dataset samples five separate traffic demands, and the average results of the policy on the five demands are used as the final performance for the dataset.

4.1.1 Dataset $D_{1\times1}$

As shown in Fig. 5a, the dataset $D_{1\times1}$ contains a typical four-leg signalized intersection common in the real world. Each approach contains four incoming lanes and three outgoing lanes, and the incoming lanes contain a left-turn lane. The length of each approach is 233 m, and the width of each lane is 3.2 m. The traffic signal in the intersection has four phases: N-S straight phase, N-S left-turn phase, E-W straight phase, and E-W left-turn phase. The right-turn phase is always allowed.

Table 1 Configurations for traffic datasets

Dataset	Number of vehicle types	Arrival rate			
		Mean	Std	Max	Min
$D_{1\times1}(1)$	1	200	19	218	172
$D_{1\times1}(2)$	1	200	105	347	55
$D_{1\times1}(3)$	1	200	102	337	53
$D_{1\times1}(4)$	1	200	29	231	153
$D_{2\times1}$	4	50	55	120	0
$D_{3\times1}$	1	225	19	246	201

$D_{1\times1}(i)$ means traffic pattern i for $D_{1\times1}$. Arrival rate is measured in terms of the number of vehicles per 120 s

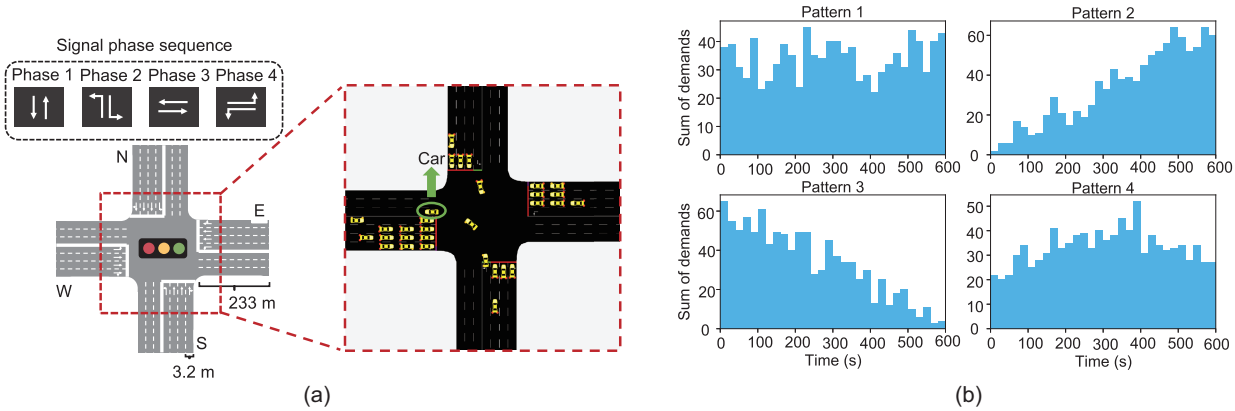

Fig. 5 The traffic environment of dataset $D_{1\times1}$: (a) structure, signal phases, and snapshot of the intersection; (b) four traffic demand patterns

In the experiment, we generate four typical traffic demand patterns (Fig. 5b). The y-axis of each subfigure indicates the number of vehicles loaded into environments at the corresponding time interval. Four traffic patterns following different distributions aim to imitate typical intersection traffic dynamics in the real world: pattern 1, following a uniform distribution, imitates smooth traffic flow during off-peak hours; patterns 2, 3, and 4, following linearly increasing, linearly decreasing, and parabolic distributions, respectively, imitate different peak traffic flow stages. Each experiment's runtime is 600 s, during which a total of 1000 vehicles are loaded into environments under one of the four traffic patterns. The vehicle turning ratio is set to 0.25.

4.1.2 Dataset $D_{2\times1}$

As shown in Fig. 6a, the dataset $D_{2\times1}$ contains a two-intersection environment with left-hand traffic. The distance between the two intersections is 100 m, which is used to verify the cooperative control performance of different policies. The dataset contains four types of vehicles, including cars, trucks, buses, and motorcycles, to simulate traffic participants in real complex traffic environments, which poses challenges for fine-grained traffic signal control. For image-based traffic signal control, the state space dimension is $64^2 \times 2 = 8192$ and the action space is $4^2 = 16$.

4.1.3 Dataset $D_{3\times1}$

The dataset is collected from real transportation systems in Dongfeng sub-district, Jinan, China (Wei et al., 2019a). As shown in Fig. 6b, the dataset $D_{3\times1}$ contains three intersections. The traffic demand is derived from the trajectory data recorded by roadside surveillance cameras. For image-based traffic signal control, the state space dimension is $64^2 \times 3 = 12\,288$ and the action space is $4^3 = 64$. The high dimensionality makes it difficult to explore effective image-based control policies.

4.2 Compared control policies

We compare DreamerV2 with two traditional rule-based traffic control policies, i.e., fixed-time and actuated polices, and three data-driven traffic control policies, i.e., deep Q-network (DQN), proximal policy optimization (PPO), and PlaNet policies. The inputs of the compared policies are position images

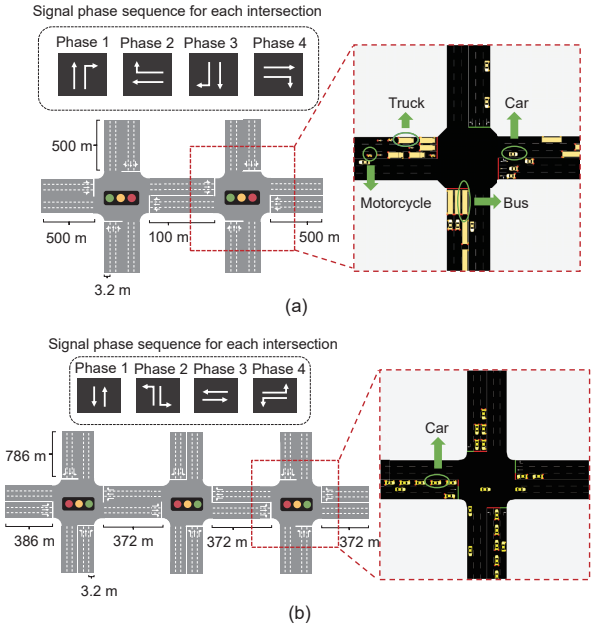

Fig. 6 The traffic environment of datasets $D_{2\times1}$ (a) and $D_{3\times1}$ (b)

or lane-level state vectors that contain the queue length and density. The formalizations of these two states are illustrated in Fig. 7. DQN and PPO are model-free, and their inputs are tested for both lane-level state vectors (denoted as Policy-V) and position images (denoted as Policy-I) for a comprehensive and fair comparison. PlaNet is model-based with the MPC architecture, and it accepts images as inputs and learns an image-based traffic model like DreamerV2. The details for the baselines and DreamerV2 are as follows:

1. Fixed-time policy: A classical fixed-time control policy called Webster's method (Webster, 1958) is used to determine the duration of green phases. Webster's method calculates the duration for minimizing the overall delay for the traffic demand. Based on Webster's method, the duration for straight phases is 30 s and that for left-turn phases is 14 s.

2. Actuated policy: The actuated controller (Newell, 1969) will switch to the next phase if detecting no vehicle via induction loop detectors related to the current phase within a sufficient time gap, which is 3 s in this study.

3. DQN-V policy: The DQN is a typical off-policy DRL approach that approximates the Q-function $Q(s,a)$ using deep neural networks (Mnih et al., 2015; Li L et al., 2016). The DQN inputs are lane-level state vectors.

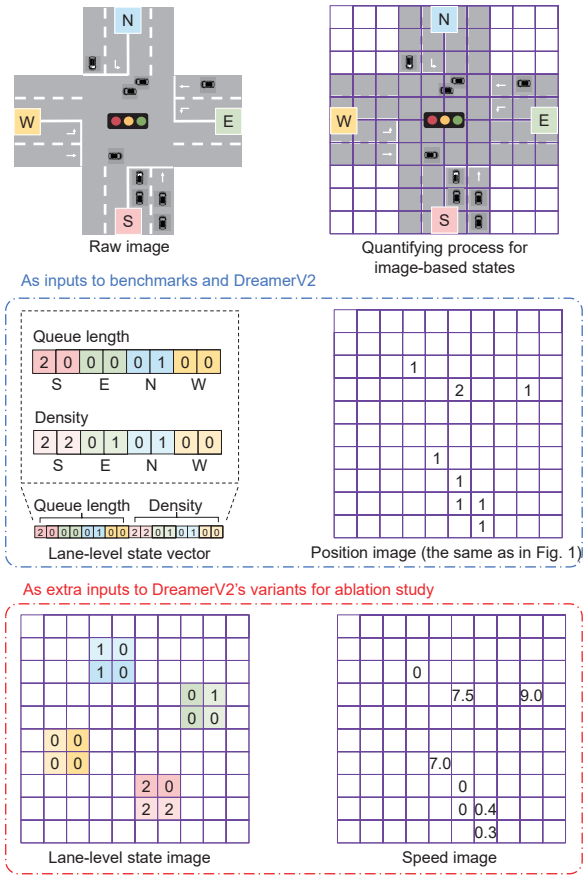

Fig. 7 Different intersection state formalizations

4. DQN-I policy: The policy is optimized via a DQN, and the DQN inputs are position images.

5. PPO-V policy: The PPO is a typical on-policy DRL approach with actor–critic learning (Mao et al., 2022). The PPO inputs are lane-level state vectors.

6. PPO-I policy: The policy is optimized via PPO, and the inputs of the policy network are position images.

7. PlaNet policy: As a model-based DRL method, PlaNet learns a world model with a latent dynamics model and a reward model and uses MPC to make decisions (Hafner et al., 2019). PlaNet's world model is similar to DreamerV2, but replaces the categorical variables with Gaussian latents. In the decision-making process, agents replan at each time step and execute the first action of the best sequence leading to the highest predicted future rewards. The inputs of PlaNet are the position images.

8. DreamerV2 policy: The applied control policy, DreamerV2 in this study, is a model-based DRL method with actor–critic learning and background planning (Hafner et al., 2022). The inputs of DreamerV2 are the position images, which are used for DQN-I, PPO-I, and PlaNet policies.

In addition to comparing the control performance of DreamerV2 and baselines, we conduct an ablation study to investigate the impact of different image-based state formalizations for DreamerV2. We also explore the feasibility of multimodal fusion for the DreamerV2 policy, as illustrated in Section 2.2. In the experiment, in addition to vehicle positions, we consider vehicle speeds and lane-level traffic states, including the queue length and density. Like vehicle position states, vehicle speed states and lane-level states are formalized as single-channel images of 64×64 separately. Their formalization processes are shown in Fig. 7. For vehicle speed images, the pixel values indicate average speeds of vehicles located in corresponding intersection positions. The approach of mapping vehicle positions in the intersection to pixel positions is shown in Section 2.2. For lane-level traffic state images, we rearrange vectors according to lane directions and combine them into the form shown in Fig. 7.

The DreamerV2 policy and its three variants based on different image-based formalizations are listed as follows:

1. DreamerV2-P (DreamerV2): The input contains only position images.

2. DreamerV2-PS: The input contains position and speed images.

3. DreamerV2-PL: The input contains position and lane-level state images.

4. DreamerV2-PSL: The input contains position, speed, and lane-level state images.

The experiments for testing the impact of different state formalizations are presented in Section 4.7.

4.3 Control performance comparison on dataset $D_{1\times1}$

As shown in Table 2, DreamerV2 outperforms the baselines in almost all metrics under four traffic patterns on the dataset $D_{1\times1}$. The policy could lead to the minimum trip delay and maximum intersection throughput, indicating that DreamerV2 could handle image-based traffic states and learn effective control policies for different traffic patterns.

A noteworthy result is that under pattern 1, DreamerV2 outperforms the fixed-time control

Table 2 Control performance comparison for different control policies on dataset $D_{1\times1}$

Traffic pattern	Performance metric	Fixed-time	Actuated	DQN-V	DQN-I	PPO-V	PPO-I	PlaNet	DreamerV2
Pattern 1	Average queue length*	2.902	4.097	4.959	8.351	3.395	4.229	4.191	**2.705**
	Average vehicle speed (m/s)	6.242	5.598	4.951	3.888	6.264	5.748	5.780	**6.623**
	Total number of output vehicles	**860**	812	793	582	829	783	756	**860**
	Average trip delay (s)	41.32	51.27	61.78	108.10	42.78	51.13	51.02	**37.76**
Pattern 2	Average queue length*	3.672	4.089	4.164	5.587	2.988	2.846	3.503	**2.512**
	Average vehicle speed (m/s)	6.129	6.410	6.349	5.411	6.971	**7.364**	6.987	7.326
	Total number of output vehicles	631	614	580	511	693	686	607	**703**
	Average trip delay (s)	51.09	52.34	55.04	75.10	40.24	39.06	47.74	**35.80**
Pattern 3	Average queue length*	7.959	7.764	5.642	5.634	4.937	5.775	6.776	**4.333**
	Average vehicle speed (m/s)	3.751	4.215	4.538	4.565	4.867	4.508	4.022	**5.326**
	Total number of output vehicles	890	**948**	862	862	897	874	866	944
	Average trip delay (s)	106.76	97.33	68.77	67.99	62.81	71.02	90.92	**56.26**
Pattern 4	Average queue length*	3.786	4.273	4.120	4.032	3.894	3.980	4.221	**2.995**
	Average vehicle speed (m/s)	5.761	5.679	5.924	5.916	6.120	6.108	6.130	**6.555**
	Total number of output vehicles	830	841	802	795	814	779	797	**858**
	Average trip delay (s)	52.37	54.85	51.99	50.29	49.01	50.83	52.73	**41.48**

* Measured by the number of vehicles. The best results are highlighted in bold

policy, while all other policies, including the data-driven policies (DQN, PPO, and PlaNet), underperform the fixed-time control policy. The results show that DreamerV2 has powerful exploration capabilities for high-performance policies because the fixed-time policy using Webster's method is a strong baseline for pattern 1. There are two reasons why the fixed-time controller has such strong performance. First, following a uniform distribution, pattern 1 is a fixed traffic pattern suitable for fixed-time control. Second, the fixed-time control policy uses Webster's method to search for the optimal duration of green phases that minimizes the vehicle delay in the isolated intersection. However, by optimizing the control policy via a world model and background planning, DeamerV2 could search for a policy with better performance than the fixed-time policy. The powerful policy exploration capability makes DreamerV2 promising for more complex traffic control scenarios.

Using the same image-based traffic states as for DreamerV2, DQN and PPO have not learned effective control policies. Table 2 shows that DQN-I has similar or worse performance compared with DQN-V, and PPO-I has similar or worse performance compared with PPO-V. These results indicate that in our settings, the two vanilla DRL approaches, DQNs and PPOs, are less capable of handling colossal exploration space induced by image inputs and learning effective policies from the image-based traffic states directly. They can handle lane-level traffic states relatively easily, which are generally formulated as vectors and adopted by most researchers in traffic signal control. However, lane-level traffic states cannot fully reflect intersection states like the image can, which restricts further improvement of the control performance for DQN-V and PPO-V. Therefore, carefully designed DQNs and PPOs are needed to cope with complicated environments. As a model-based DRL policy, PlaNet performs worse than DreamerV2. The result indicates that the DreamerV2's world model using categorical variables and background-planning-based optimization mechanism is more effective for image-based traffic signal control.

To better understand the performance change in the control policy during training, we visualize the episode reward curve in evaluation over environment steps for different control policies. As shown in Fig. 8, DreamerV2 can learn a high-performance control policy using only a small number of environment samples, whereas model-free DRL policies need more environment data to converge. This result validates that the model-based DreamerV2 has higher data efficiency than the model-free DRL approaches. DreamerV2 can generate large amounts of data for training through its learned world model, whereas DQNs and PPOs can learn only policies by samples from the environment. High data efficiency makes DreamerV2 suitable for scaling to real-world traffic

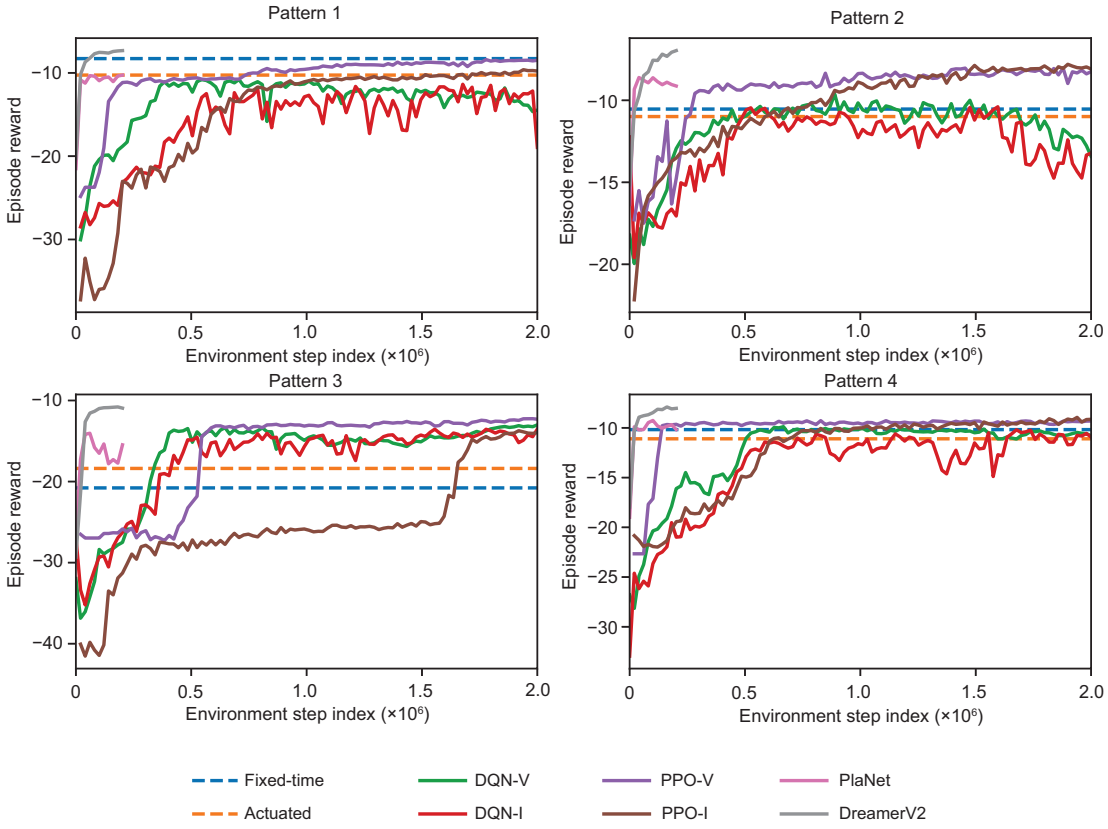

Fig. 8 Evaluation rewards over environment steps on dataset $D_{1\times1}$ (References to color refer to the online version of this figure)

signal control problems, because real-world samples are expensive, especially for extreme scenarios.

4.4 Control performance comparison on datasets $D_{2\times1}$ and $D_{3\times1}$

The results of the two experiments are shown in Table 3. DreamerV2 achieves the optimal control performance for both scenarios.

In the dataset $D_{2\times1}$, the DreamerV2 policy achieves the best performance in all metrics. Moreover, an interesting result is that the image-based traffic control approach shows an overwhelming advantage in complex intersection scenarios containing heterogeneous vehicles with different types. Specifically, image-based DQN-I and PPO-I outperform vector-based DQN-V and PPO-V, respectively. This indicates that in complex intersection scenarios with heterogeneous vehicle types, image-based traffic control can better model the relationships between vehicles and thus achieve better control performance.

In the dataset $D_{3\times1}$, DreamerV2 also achieves the best performance in all metrics. In addition, the figure of the evaluation reward curve (Fig. 9) illustrates that DreamerV2 has the highest data utilization, and the near-optimal control policy can be explored using data of 2×10^5 environment steps.

4.5 Decision time comparison

Fig. 10 shows the decision time statistics for different policies in the test environment. All policies are tested on the same Linux server (CPU: Intel Xeon CPU E5-2650 v4 @2.20 GHz, GPU: NVIDIA TITAN V). We find that the DreamerV2 decision time is similar to that of the model-free DRL policies DQN and PPO. Their average decision time per step is $<$0.007 s. DreamerV2 could guarantee real-time decision-making because its decision module can be detached from the prediction module in the execution stage. In contrast, MPC-based PlaNet needs planning during decision-making, so it takes longer

Table 3 Control performance comparison for different control policies on datasets $D_{2\times1}$ and $D_{3\times1}$

Dataset	Performance metric	Fixed-time	Actuated	DQN-V	DQN-I	PPO-V	PPO-I	PlaNet	DreamerV2
$D_{2\times1}$	Average queue length*	1.194	0.791	2.190	1.091	0.923	0.792	0.810	**0.518**
	Average vehicle speed (m/s)	6.535	6.908	4.281	5.635	5.537	6.420	6.744	**7.318**
	Total number of output vehicles	**250**	**250**	196	233	237	247	249	**250**
	Average trip delay (s)	97.57	72.63	151.46	93.27	82.90	76.66	79.03	**58.19**
$D_{3\times1}$	Average queue length*	2.662	1.462	2.091	2.394	1.326	1.238	1.711	**1.093**
	Average vehicle speed (m/s)	5.952	6.306	6.116	6.067	6.259	6.300	6.114	**6.332**
	Total number of output vehicles	648	699	663	654	681	691	674	**704**
	Average trip delay (s)	122.82	112.38	117.48	120.46	113.94	112.06	117.83	**110.79**

* Measured by the number of vehicles. The best results are highlighted in bold

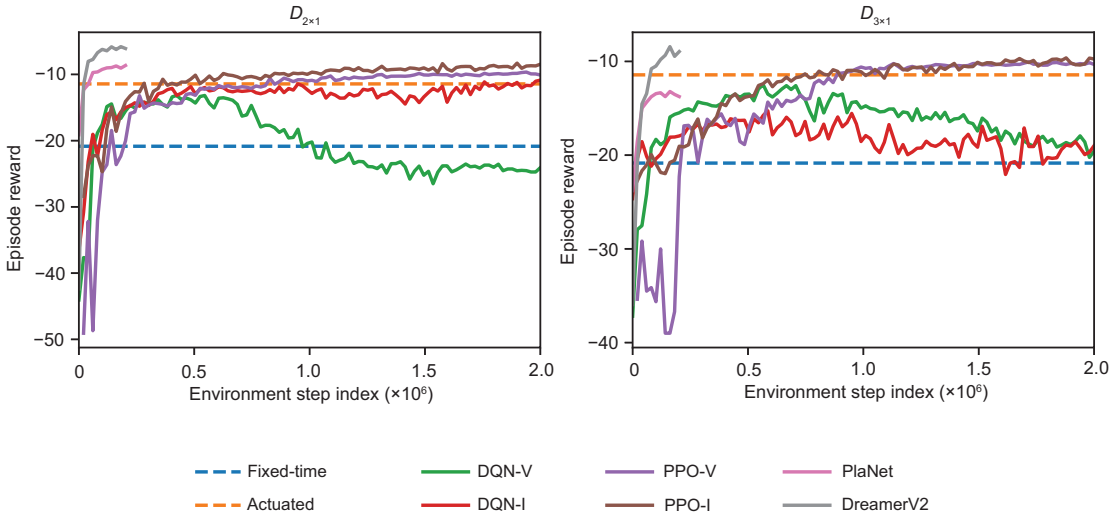

Fig. 9 Evaluation rewards over environment steps on datasets $D_{2\times1}$ and $D_{3\times1}$ (References to color refer to the online version of this figure)

to make decisions than other policies. In the experiment, the average decision time of PlaNet is 0.058 s, eight times longer than that of DreamerV2. Moreover, the decision time for PlaNet will increase for complex environments with higher-dimensional image inputs, making the handling of real-time traffic control scenarios difficult. Unlike PlaNet, DreamerV2 does not require prediction during decision-making. So, its decision time is linearly related to the dimensionality of the image, making it ideal for handling complex traffic dynamics.

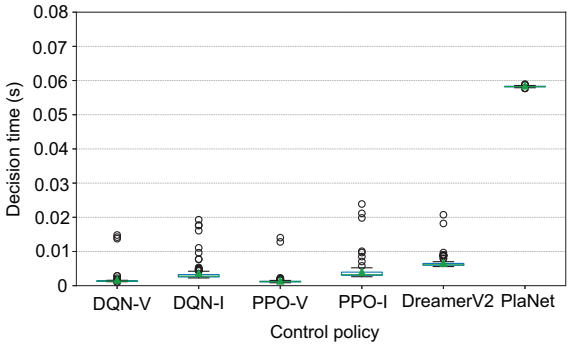

Fig. 10 Decision time for different policies

4.6 Visualization of traffic prediction

To evaluate the prediction ability of DreamerV2, we visualize its predicted image-based traffic states and check the prediction error. Here, we use the DreamerV2 policy optimized on pattern 1. Fig. 11 shows the DreamerV2 prediction of traffic states for the next 10 steps (50 s) based on the data from the past 5 steps. We also plot the ground truth of future traffic states and the corresponding prediction error. We find that DreamerV2 can accurately predict the near-future image-based traffic state by

relying on specially designed predictive learning. The predicted position changes because all vehicles are consistent with the actual changes. The prediction results imply that DreamerV2 with a traffic world model can accurately describe and forecast the image-based traffic dynamics. The effective image-based traffic modeling provides interpretability for DreamerV2 decisions.

Although the traffic world model of DreamerV2 shows a strong prediction ability for traffic dynamics following the DreamerV2 policy, its generalization in other control policies needs further validation. Consequently, we conduct experiments to test the traffic prediction performance of the world model for two kinds of traffic dynamics that follow opposite signal control policies: N-S green policy and E-W green policy. The N-S green policy and the E-W green policy maintain phase 1 (N-S straight) and phase 3 (E-W straight), respectively, in any traffic states. The multi-step prediction for traffic dynamics following the two policies is shown in Fig. 12. To compare the prediction results using the two policies with those using the DreamerV2 policy, we first feed the traffic world model with the same five-step historical traffic states and actions as in Fig. 11. Then, the world model uses sequences of future actions (corresponding to phase indices) for the N-S green policy and E-W green policy, respectively, to predict traffic dynamics following the two opposite control policies.

Fig. 12 shows that even given the same historical traffic states, the world model is still able to predict different traffic state changes based on different sequences of future actions. The world model can predict smooth traffic flow in N-S lanes and traffic congestion in E-W lanes for the N-S green policy, but makes the opposite prediction for the E-W green policy. We can also find that for traffic prediction under different control policies, the difference of

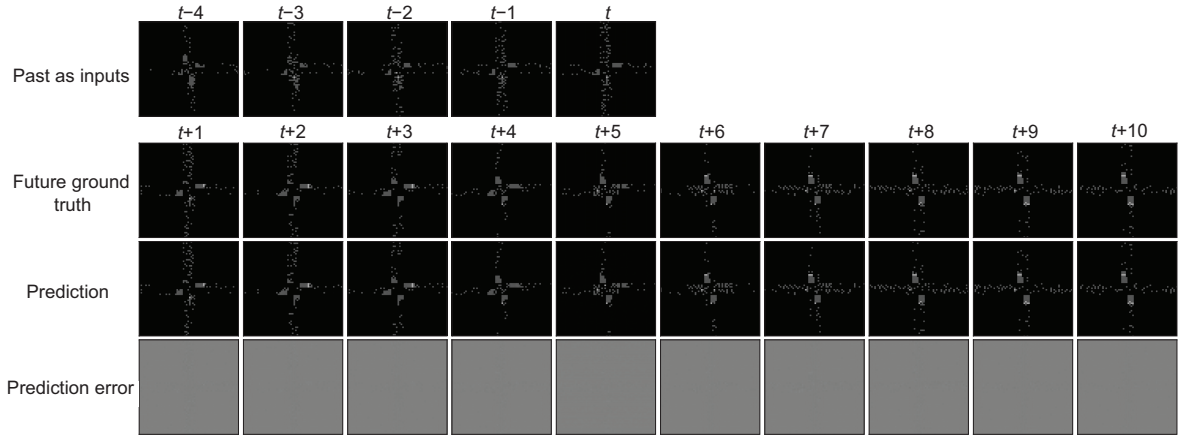

Fig. 11 Visualization of the future traffic states (ground truth) and the predicted traffic states

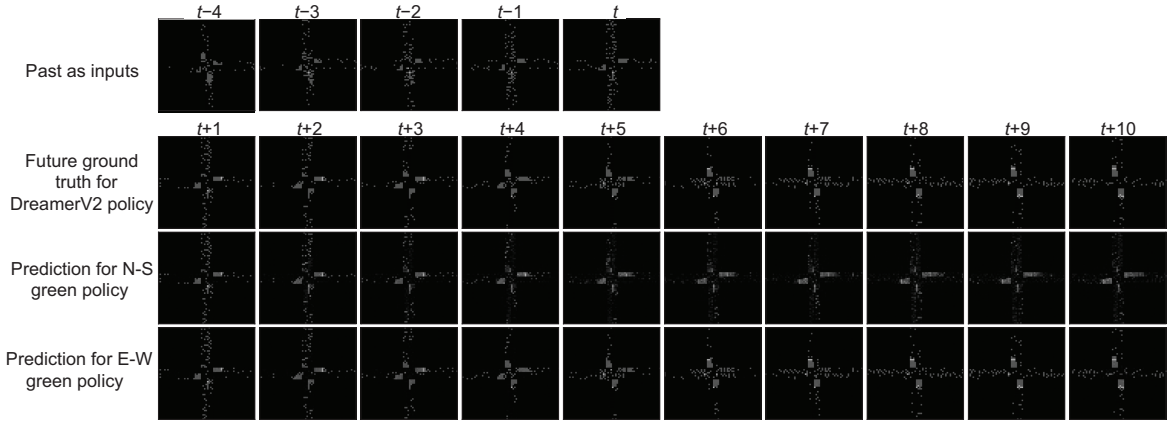

Fig. 12 Visualization of world model predictions of traffic states under different control policies

future states increases with the growth of prediction length. The prediction for the N-S green policy and E-W green policy shows a significant difference after time step $t+3$. Unfortunately, the world model does not accurately predict changes in traffic dynamics under control policies other than DreamerV2: some blurred pixels in the predicted image-based traffic states are not representations of vehicles. Although the learned world model cannot make fine-grained forecasts for all traffic dynamics in current settings, its ability to predict traffic trends makes predictors promising to supplement traffic simulators or mathematical models. We can view traffic evolution under different policies via the image-based world model.

4.7 Impact of different state formalizations

As shown in Table 4, DreamerV2-PL using position and lane-level state images achieves the best performance in most metrics. This result indicates that DreamerV2 can benefit from lane-level information about queue length and density. Although the position image implicitly contains lane-level information, the feature may not be fully captured by DreamerV2-P. So, multimodal fusion is necessary as illustrated in Section 2.2. Furthermore, the multimodal features used for fusion need to be carefully chosen. We find that the speed image does not improve the control performance of DreamerV2-PS compared with DreamerV2-P. This might be because DreamerV2 can implicitly extract speed information from consecutive position images based on its GRU modules, which can track long-term dependency.

5 Conclusions

This paper analyzes the importance of prediction for traffic signal control and reviews the typical model-based control frameworks. Although model-based traffic signal control has been studied for many years, some problems remain to be solved. The problems lie in the balance between the effectiveness and timeliness of traffic signal control under complex traffic dynamics as well as the integrated approach to prediction and control.

To tackle these problems, we use an image containing vehicle positions at the intersection to describe traffic dynamics and introduce DreamerV2 to learn an image-based traffic world model, which performs background planning to help optimize the traffic signal control policy. The learned control policy could make decisions independent of the world model's predictor, enabling effective and real-time traffic signal control. We can also use the traffic world model to derive future traffic trends under the current policy or other policies on demand. This mechanism for flexible use of the world model provides a new perspective for building a real-world-oriented traffic signal control framework outside the mainstream MPC framework. In the execution stage, the MPC framework using planning at

Table 4 Control performance comparison for different state formalizations on dataset $D_{1\times1}$

Traffic pattern	Performance metric	DreamerV2-P	DreamerV2-PS	DreamerV2-PL	DreamerV2-PSL
Pattern 1	Average queue length*	2.705	2.949	**2.583**	2.608
	Average vehicle speed (m/s)	6.623	6.519	**6.801**	6.766
	Total number of output vehicles	860	867	865	**875**
	Average trip delay (s)	37.76	39.70	**36.07**	36.87
Pattern 2	Average queue length*	2.512	2.684	**2.460**	3.049
	Average vehicle speed (m/s)	7.326	7.151	**7.464**	6.721
	Total number of output vehicles	703	718	**723**	695
	Average trip delay (s)	35.80	37.24	**35.30**	40.51
Pattern 3	Average queue length*	4.333	4.903	**4.012**	5.324
	Average vehicle speed (m/s)	5.326	5.004	**5.599**	4.830
	Total number of output vehicles	944	921	**964**	914
	Average trip delay (s)	56.26	61.39	**53.15**	65.92
Pattern 4	Average queue length*	2.995	3.284	**2.892**	3.347
	Average vehicle speed (m/s)	6.555	6.376	**6.693**	6.410
	Total number of output vehicles	858	850	**869**	836
	Average trip delay (s)	41.48	43.67	**40.26**	44.67

* Measured by the number of vehicles. The best results are highlighted in bold

decision time leads to difficulties in real-time decision-making. In contrast, world-model-based decision-making does not depend on the prediction, but we can make accurate multi-step predictions on demand to estimate the controller's performance.

To better apply the traffic world model to the futuristic real-world traffic signal control, we need to further validate its feasibility in more complex traffic scenarios. First, it is necessary to use a unified traffic control policy incorporating multimodal fusion to handle a wide variety of traffic environments, including different traffic patterns, road conditions, and other factors such as weather. Second, we need to consider networked traffic control. It is feasible to study how to build an effective multi-intersection world model. The model can perform city-wide traffic prediction (Dai et al., 2019; Li ZS et al., 2022), and help optimize coordinated control policies by introducing multi-agent RL (Bertsekas, 2021; Zhang KQ et al., 2021) or hierarchical RL (Liu et al., 2021).

For future applications of the traffic world model, we consider integrating it with current advanced traffic control systems, like human-in-the-loop recommendation systems (Jin et al., 2021), to explain the recommended policy by intelligent machines. The world model also provides a solution to constructing artificial systems of parallel transportation systems. We can simulate different traffic evolutions in the world model for many computational experiments. In a sense, the traffic world model enriches the study of artificial systems by enabling them to take on forms other than mathematical and simulation models. By optimizing the control policy via predictive learning in the parallel learning framework (Li L et al., 2017), the world model can facilitate parallel execution and prescribe the operation of real-world transportation systems.

Contributors

Xingyuan DAI and Fei-Yue WANG designed the research. Xiao WANG and Yisheng LV contributed ideas for experiments and analysis. Chen ZHAO created the simulation platform. Xingyuan DAI and Yilun LIN performed simulations and analysis. Fei-Yue WANG managed the project. Xingyuan DAI and Chen ZHAO drafted the paper. Xiao WANG, Yisheng LV, and Fei-Yue WANG revised and finalized the paper.

Compliance with ethics guidelines

Xingyuan DAI, Chen ZHAO, Xiao WANG, Yisheng LV, Yilun LIN, and Fei-Yue WANG declare that they have no conflict of interest.

References

Abdoos M, Bazzan ALC, 2021. Hierarchical traffic signal optimization using reinforcement learning and traffic prediction with long-short term memory. *Expert Syst Appl*, 171:114580. https://doi.org/10.1016/j.eswa.2021.114580

Bertsekas D, 2021. Multiagent reinforcement learning: rollout and policy iteration. *IEEE/CAA J Autom Sin*, 8(2):249-272. https://doi.org/10.1109/JAS.2021.1003814

Dai XY, Fu R, Zhao EM, et al., 2019. DeepTrend 2.0: a light-weighted multi-scale traffic prediction model using detrending. *Transp Res Part C Emerg Technol*, 103:142-157. https://doi.org/10.1016/j.trc.2019.03.022

Guo QQ, Li L, Ban XG, 2019. Urban traffic signal control with connected and automated vehicles: a survey. *Transp Res Part C Emerg Technol*, 101:313-334. https://doi.org/10.1016/j.trc.2019.01.026

Hafner D, Lillicrap T, Fischer I, et al., 2019. Learning latent dynamics for planning from pixels. Proc 36th Int Conf on Machine Learning, p.2555-2565.

Hafner D, Lillicrap TP, Norouzi M, et al., 2022. Mastering Atari with discrete world models. https://arxiv.org/abs/2010.02193

Hao ZZ, Boel R, Li ZW, 2018. Model based urban traffic control, part I: local model and local model predictive controllers. *Transp Res Part C Emerg Technol*, 97:61-81. https://doi.org/10.1016/j.trc.2018.09.026

Jin JC, Guo HF, Xu J, et al., 2021. An end-to-end recommendation system for urban traffic controls and management under a parallel learning framework. *IEEE Trans Intell Transp Syst*, 22(3):1616-1626. https://doi.org/10.1109/TITS.2020.2973736

Kim D, Jeong O, 2019. Cooperative traffic signal control with traffic flow prediction in multi-intersection. *Sensors*, 20(1):137. https://doi.org/10.3390/s20010137

Li L, Lv YS, Wang FY, 2016. Traffic signal timing via deep reinforcement learning. *IEEE/CAA J Autom Sin*, 3(3):247-254. https://doi.org/10.1109/JAS.2016.7508798

Li L, Lin YL, Zheng NN, et al., 2017. Parallel learning: a perspective and a framework. *IEEE/CAA J Autom Sin*, 4(3):389-395. https://doi.org/10.1109/JAS.2017.7510493

Li ZS, Xiong G, Tian YL, et al., 2022. A multi-stream feature fusion approach for traffic prediction. *IEEE Trans Intell Transp Syst*, 23(2):1456-1466. https://doi.org/10.1109/TITS.2020.3026836

Liang XY, Du XS, Wang GL, et al., 2019. A deep reinforcement learning network for traffic light cycle control. *IEEE Trans Veh Technol*, 68(2):1243-1253. https://doi.org/10.1109/TVT.2018.2890726

Liu CH, Zhu F, Liu Q, et al., 2021. Hierarchical reinforcement learning with automatic sub-goal identification. *IEEE/CAA J Autom Sin*, 8(10):1686-1696. https://doi.org/10.1109/JAS.2021.1004141

Lopez PA, Behrisch M, Bieker-Walz L, et al., 2018. Microscopic traffic simulation using SUMO. Proc 21st IEEE Int Conf on Intelligent Transportation Systems, p.2575-2582. https://doi.org/10.1109/ITSC.2018.8569938

Lv YS, Duan YJ, Kang WW, et al., 2014. Traffic flow prediction with big data: a deep learning approach. *IEEE Trans Intell Transp Syst*, 16(2):865-873. https://doi.org/10.1109/TITS.2014.2345663

Mao F, Li ZH, Li L, 2022. A comparison of deep reinforcement learning models for isolated traffic signal control. *IEEE Intell Transp Syst Mag*, early access. https://doi.org/10.1109/MITS.2022.3144797

Mei ZY, Tan Z, Zhang W, et al., 2019. Simulation analysis of traffic signal control and transit signal priority strategies under arterial coordination conditions. *Simulation*, 95(1):51-64. https://doi.org/10.1177/0037549718757651

Mnih V, Kavukcuoglu K, Silver D, et al., 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529-533. https://doi.org/10.1038/nature14236

Newell GF, 1969. Properties of vehicle-actuated signals: I. one-way streets. *Transp Sci*, 3(1):30-52.

Nie J, Yan J, Yin HL, et al., 2021. A multimodality fusion deep neural network and safety test strategy for intelligent vehicles. *IEEE Trans Intell Veh*, 6(2):310-322. https://doi.org/10.1109/TIV.2020.3027319

Seng D, Lv FS, Liang ZY, et al., 2021. Forecasting traffic flows in irregular regions with multi-graph convolutional network and gated recurrent unit. *Front Inform Technol Electron Eng*, 22(9):1179-1193. https://doi.org/10.1631/FITEE.2000243

Sutton RS, Barto AG, 2018. Reinforcement Learning: an Introduction (2nd Ed.). The MIT Press, Cambridge, USA.

Varaiya P, 2013. Max pressure control of a network of signalized intersections. *Transp Res Part C Emerg Technol*, 36:177-195. https://doi.org/10.1016/j.trc.2013.08.014

Wang FY, 2010. Parallel control and management for intelligent transportation systems: concepts, architectures, and applications. *IEEE Trans Intell Transp Syst*, 11(3):630-638. https://doi.org/10.1109/TITS.2010.2060218

Wang HN, Liu N, Zhang YY, et al., 2020. Deep reinforcement learning: a survey. *Front Inform Technol Electron Eng*, 21(12):1726-1744. https://doi.org/10.1631/FITEE.1900533

Wang J, Li R, Wang J, et al., 2020. Artificial intelligence and wireless communications. *Front Inform Technol Electron Eng*, 21(10):1413-1425. https://doi.org/10.1631/FITEE.1900527

Webster FV, 1958. Traffic Signal Settings. Technical Report No. 39, Road Research Laboratory, UK.

Wei H, Xu N, Zhang HC, et al., 2019a. CoLight: learning network-level cooperation for traffic signal control. Proc 28th ACM Int Conf on Information and Knowledge Management, p.1913-1922. https://doi.org/10.1145/3357384.3357902

Wei H, Chen CC, Zheng GJ, et al., 2019b. PressLight: learning max pressure control to coordinate traffic signals in arterial network. Proc 25th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining, p.1290-1298. https://doi.org/10.1145/3292500.3330949

Wiering M, 2000. Multi-agent reinforcement learning for traffic light control. Proc 17th Int Conf on Machine Learning, p.1151-1158.

Xiao Y, Codevilla F, Gurram A, et al., 2022. Multimodal end-to-end autonomous driving. *IEEE Trans Intell Transp Syst*, 23(1):537-547. https://doi.org/10.1109/TITS.2020.3013234

Xiong G, Dong XS, Lu H, et al., 2020. Research progress of parallel control and management. *IEEE/CAA J Autom Sin*, 7(2):355-367. https://doi.org/10.1109/JAS.2019.1911792

Ye BL, Wu WM, Ruan KY, et al., 2019. A survey of model predictive control methods for traffic signal control. *IEEE/CAA J Autom Sin*, 6(3):623-640. https://doi.org/10.1109/JAS.2019.1911471

Yu ZX, Liang SX, Wei L, et al., 2020. MaCAR: urban traffic light control via active multi-agent communication and action rectification. Proc 29th Int Joint Conf on Artificial Intelligence, p.2491-2497. https://doi.org/10.24963/ijcai.2020/345

Zhang HC, Kafouros M, Yu Y, 2020. PlanLight: learning to optimize traffic signal control with planning and iterative policy improvement. *IEEE Access*, 8:219244-219255. https://doi.org/10.1109/ACCESS.2020.3041441

Zhang KQ, Yang ZR, Basar T, 2021. Decentralized multi-agent reinforcement learning with networked agents: recent advances. *Front Inform Technol Electron Eng*, 22(6):802-814. https://doi.org/10.1631/FITEE.1900661

Zhao YF, Gao H, Wang S, et al., 2017. A novel approach for traffic signal control: a recommendation perspective. *IEEE Intell Transp Syst Mag*, 9(3):127-135. https://doi.org/10.1109/MITS.2017.2709779

Zhu FH, Lv YS, Chen YY, et al., 2020. Parallel transportation systems: toward IoT-enabled smart urban traffic control and management. *IEEE Trans Intell Transp Syst*, 21(10):4063-4071. https://doi.org/10.1109/TITS.2019.2934991